Visualization and Analysis of Co-occurrence and Cross-tabulation data in Medical Research

Joseph I. Bormel, M.D., UCLA Center for Health Sciences
Linda R. Ferguson, Ph.D., UCLA Office of Academic Computing

ABSTRACT

Analyzing raw data can be prohibitively time consuming. A variety of graphical techniques have been developed to address this problem. Although graphical analysis can provide a simple yet comprehensive overview of a large dataset, often these techniques fail to capture the essence of data trends. In addition, the ability to easily query any component of the data subset frequently remains burdensome. In this paper, we present a general method to address these issues for cross-tabulation tables and provide examples of their use in medical research.

INTRODUCTION

Graphical display of information in medicine presents a variety of challenges[1]. For many applications, simple numerical tables of frequencies provides the first step in analyzing the patterns in the data set. Often, the next step in looking for patterns is summarizing all observations in a database along two axes to form a cross-tabulation or co-occurrence table. Frequently, these methods of viewing data fail to reveal important trends.

Recognizing significant patterns of disease progression in a large patient population over time is a task which demonstrates these difficulties. We recently studied a cohort of 410 patients from Cooperative Systematic Studies of Rheumatic Diseases (CSSRD) clinics[2]. These patients were entered from ten university-based rheumatology clinics in a prospective study within one year of onset of connective tissue disease (CTD). The purpose of the study was to gain a better understanding of progression or remission of undifferentiated connective tissue diseases (UCTD). Correlation between disease features at entry and subsequent remission were examined to identify features at entry that predict remission. These 410 patients were divided initially into 18 disease subgroups which over time redistributed into 25 disease subgroups. We identified disease progression via transition paths between disease subgroups, which show the number of patients who eventually develop another disease or enter remission versus those who remain in the same disease subgroup.

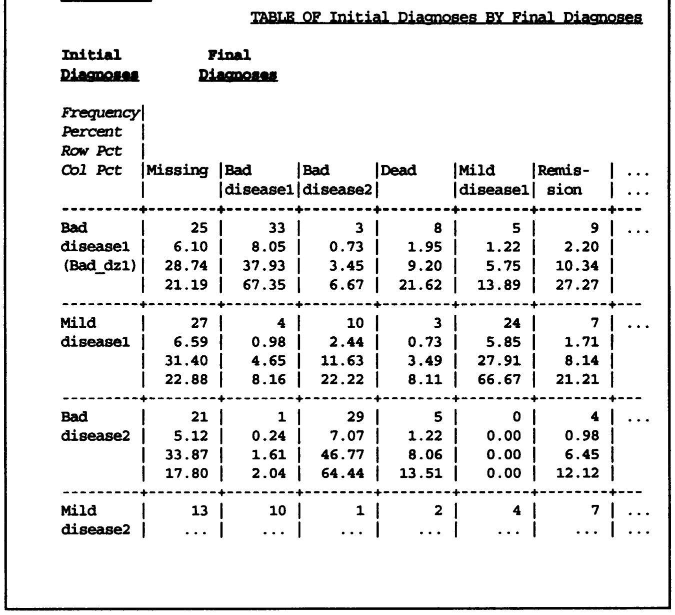
We developed a graphical exploratory method to find the important relationships in the data, as well as to facilitate more in-depth examination. First, we used a statistical program's procedure (SAS's proc freq, a frequency determining procedure) to produce a cross-tabulation table to show *Inital Diagnoses* and *Final Diagnoses*. This table showed the number of patients with each disease at entry into the study (initial diagnosis) and which disease category they were in at the last examination (final diagnosis). An example of such a table is shown in **table 1**. As described in this paper, we then developed a Visual Basic program to generate a graphical display of the data which supported further analysis on an interactive basis. This method is applicable to all forms of cross-tabulation tables, whether they arise from statistical packages as ours did, or relational database models of computerized patient records.

BACKGROUND

Exploratory Data Analysis (EDA) software facilitates unstructured, iterative visual exploration of relationships in complex datasets with the aid of multiple graphical displays linked to the data's information content. EDA methods can be essential in analyses of data sets because they can provide perspectives which would otherwise be obscured or completely hidden, especially when the data volume and variation contained is massive. EDA methods have been used in conjunction with statistical methods to build prediction models in medicine and psychiatry[3,4]. In addition to proving useful in enabling study of data from multiple perspectives and uncovering elusive data trends, it is also a powerful communications tool in the subsequent presentation of these findings.

To successfully view large amounts of clinical data, it is necessary to integrate a database management system, graphical

Table 1



```
                      TABLE OF Initial Diagnoses BY Final Diagnoses

Initial        Final
Diagnoses      Diagnoses

Frequency|
Percent  |
Row Pct  |
Col Pct  |Missing |Bad     |Bad     |Dead    |Mild    |Remis-  | ...
         |        |disease1|disease2|        |disease1| sion   | ...
---------+--------+--------+--------+--------+--------+--------+---
Bad      |     25 |     33 |      3 |      8 |      5 |      9 | ...
disease1 |   6.10 |   8.05 |   0.73 |   1.95 |   1.22 |   2.20 |
(Bad_dz1)|  28.74 |  37.93 |   3.45 |   9.20 |   5.75 |  10.34 |
         |  21.19 |  67.35 |   6.67 |  21.62 |  13.89 |  27.27 |
---------+--------+--------+--------+--------+--------+--------+---
Mild     |     27 |      4 |     10 |      3 |     24 |      7 | ...
disease1 |   6.59 |   0.98 |   2.44 |   0.73 |   5.85 |   1.71 |
         |  31.40 |   4.65 |  11.63 |   3.49 |  27.91 |   8.14 |
         |  22.88 |   8.16 |  22.22 |   8.11 |  66.67 |  21.21 |
---------+--------+--------+--------+--------+--------+--------+---
Bad      |     21 |      1 |     29 |      5 |      0 |      4 | ...
disease2 |   5.12 |   0.24 |   7.07 |   1.22 |   0.00 |   0.98 |
         |  33.87 |   1.61 |  46.77 |   8.06 |   0.00 |   6.45 |
         |  17.80 |   2.04 |  64.44 |  13.51 |   0.00 |  12.12 |
---------+--------+--------+--------+--------+--------+--------+---
Mild     |     13 |     10 |      1 |      2 |      4 |      7 | ...
disease2 |    ... |    ... |    ... |    ... |    ... |    ... | ...
```

0195-4210/94/$5.00 © 1994 AMIA, Inc.

display and statistical software. [5] We were not aware of any program that would integrate with the SAS system to perform the interactive graphical EDA techniques necessary. We therefore developed the Visual Basic program described in this paper.

METHODS

Our initial representation of our study population's disease course, shown in **table 1** was produced using the SAS PROC FREQ procedure. The table shows initial visit diagnosis by final visit diagnosis in an abbreviated fashion and using generalized labels (e.g. *Bad disease1* instead of the actual disease name). All possible courses a patient could take are represented by cells in this table. The full table included 18 categories of initial diseases (shown in the rows) and 25 categories of disease at the completion of the study (shown in the columns). In this sample table, the first cell containing 25 represents 25 patients with an initial diagnosis of 'Bad disease1' whose final diagnosis state was 'Missing'. Similarly, the next cell down containing 27 represents patients who initially had 'Mild disease1' whose final diagnosis was 'Missing'. This table did not convey a clear sense of our data due to the large number of cells, size disparity between adjacent rows and columns, sparseness and separation over multiple pages. Thus, it was difficult to identify the significant transition paths that link patients' initial and final diagnoses.

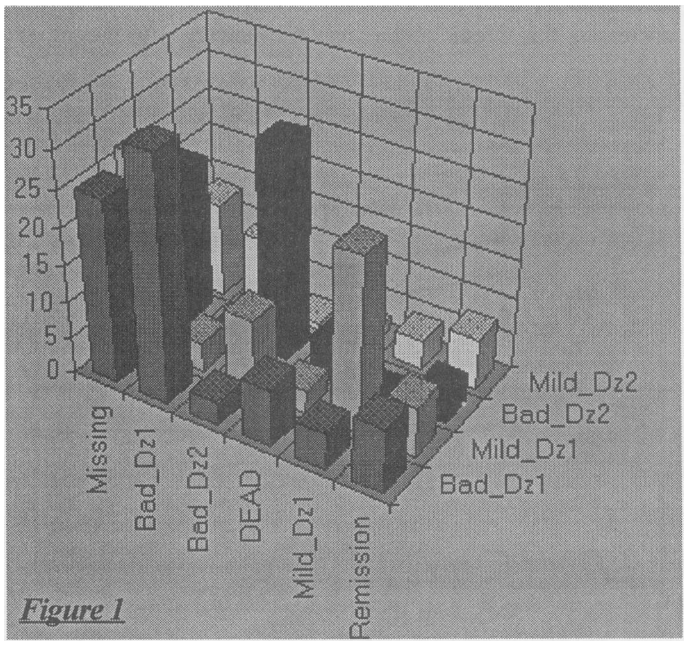


Figure 1

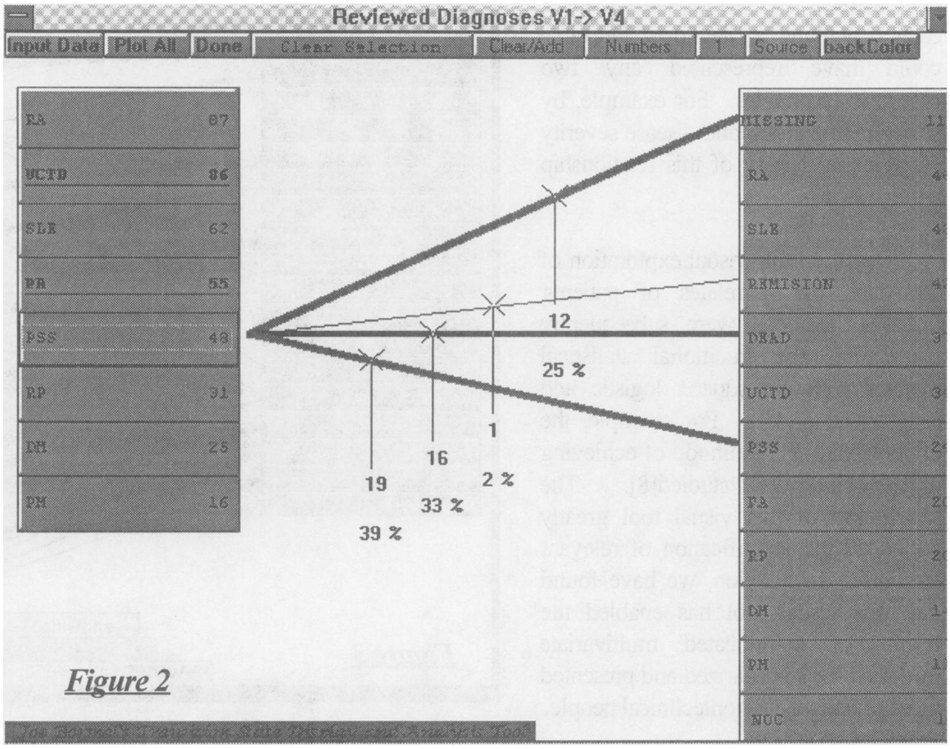


Figure 2

To facilitate interpretation, the next step was to convert the table into a graphical format. This can be done in many ways using a variety of graphing tools and techniques[6].

A graphical depiction of **table 1** using a three dimensional model (created with Microsoft Excel, in **figure 1**) was only a slight improvement. It did not communicate the flow of patients between subgroups. To overcome this, a pilot program was written to graphically represent the number of patients following each transition path. This tool was subsequently extended in Visual Basic to allow interactive query of patient subpopulations, by user subgroup selection and automatic generation of procedure code to statistically analyze those subgroups.

A sample display produced by this Visual Basic program is shown in **figure 2**. The initial disease selected in this case is PSS (Progressive Systemic Sclerosis)[7]. It was selected by pointing to the button containing the legend "PSS" with a mouse and selecting it with a mouse button press. This caused the distribution of these patients' disease courses to be displayed. For example, the top line, connecting PSS on the left to MISSING on the right, represents 12 patients, or 25% of all of the 48 patients initially diagnosed with the disease PSS. Any patient subgroup can be selected in this way, with the specific disease progression pattern instantly displayed. The interactive nature of this process combined with the clearing of data which is unrelated to the currently chosen subgroup helps to selectively focus the user 's attention.

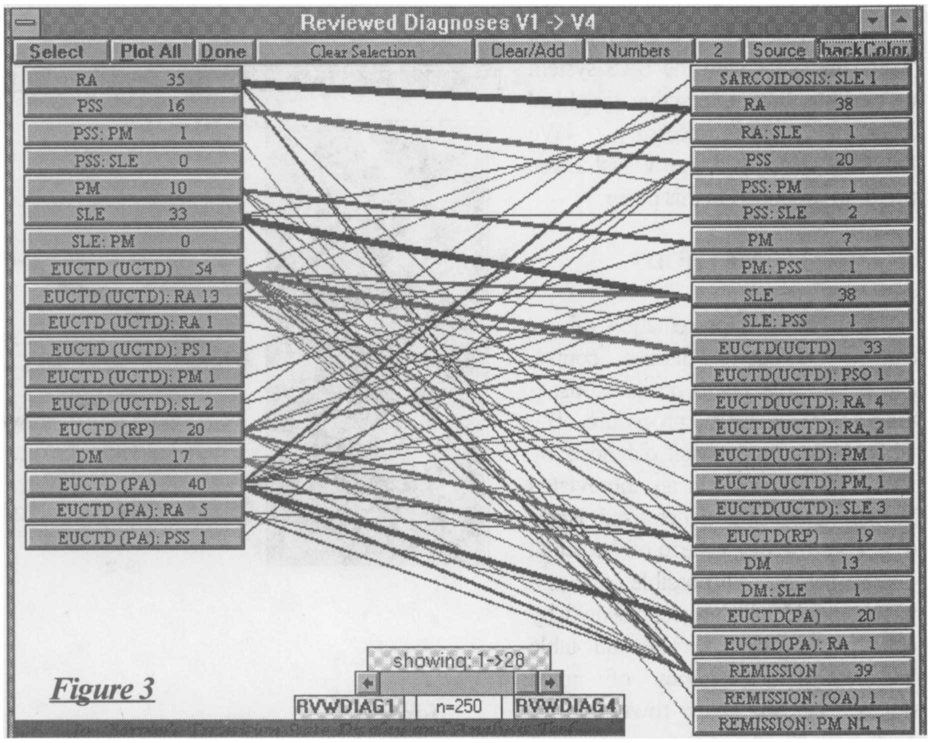


Figure 3

Although this example demonstrates two categorical variables, each containing a diagnosis, the table could have represented any two categorical variables. For example, by selecting age group and disease severity variables, a display of this relationship would result.

Based on this visual exploration of the data, the correlates of patients' disease transitions were subsequently studied in the traditional statistical manner with chi square, logistic and categorical models. For example, the effect of age on likelihood of achieving a remission was studied[8]. The availability of this visual tool greatly facilitated the identification of relevant variables. In addition, we have found that this visual tool has enabled the results of complicated multivariate models to be summarized and presented more effectively to nontechnical people.

SYSTEM DESIGN

Implementing this method involved the use of two programming languages, Visual Basic and PERL[9]. Visual Basic is a commercial programming language sold by Microsoft Corporation. It is well suited to building graphical user interfaces and to quickly developing simple programs. There are many books and articles which describe programming in this language.

The second language, Perl, was developed by Larry Wall in 1986. It officially stands for Practical Extraction and Report Language and is ideal for extracting data from files and reformatting the data for other purposes. It is distributed with source code and is available at no charge. In the application described above, a Perl program translated the SAS PROC FREQ output into data more easily usable by the Visual Basic code.

In the course of developing the visualization engine and enhancing its interactive capabilities, it became clear that SAS's data subsetting and analysis capabilities and SAS's expressivity were fundamental components for powerful interactive analysis. For these reasons, we have been developing a generalized graphical front-end for data analysis which is specifically designed to facilitate analyses of disease transition in populations over time. The user can interactively select variables contained in an application's tables, and the Visual Basic program will graphically depict the relationships.

Using this approach, co-occurrence data containing large numbers of subsets can be quickly examined. For example, **figure 3** summarizes the disease course of the 250 non-missing patients in the earlier mentioned study. All transition paths, even those containing only one patient is shown. Thicker lines represent larger numbers of patients. Because of the density of information, this display is of limited use. However, some large patterns are evident. Also, on a color display, the display is slightly more useful.

To exclude subsets that occurred less frequently, a threshold can be adjusted to allow the user to focus on the larger groups. This signficantly reduces the cluttered appearance. The result is shown in **figure 4** which contains the same data as **figure 3**, but excludes all groups with less than 8 occurrences. By successively increasing this threshold, the interface allows you to decompose

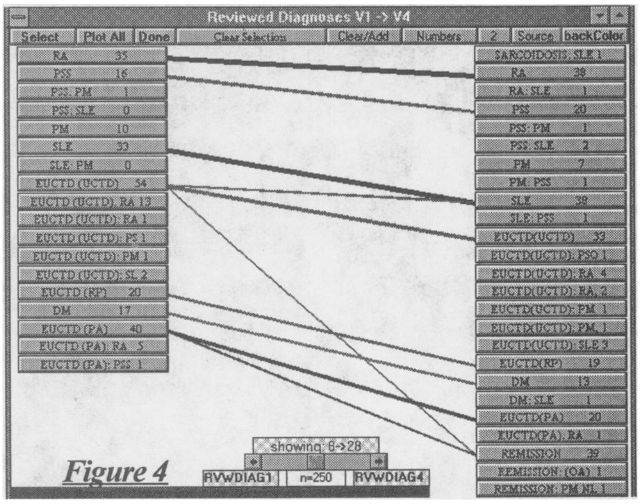


Figure 4

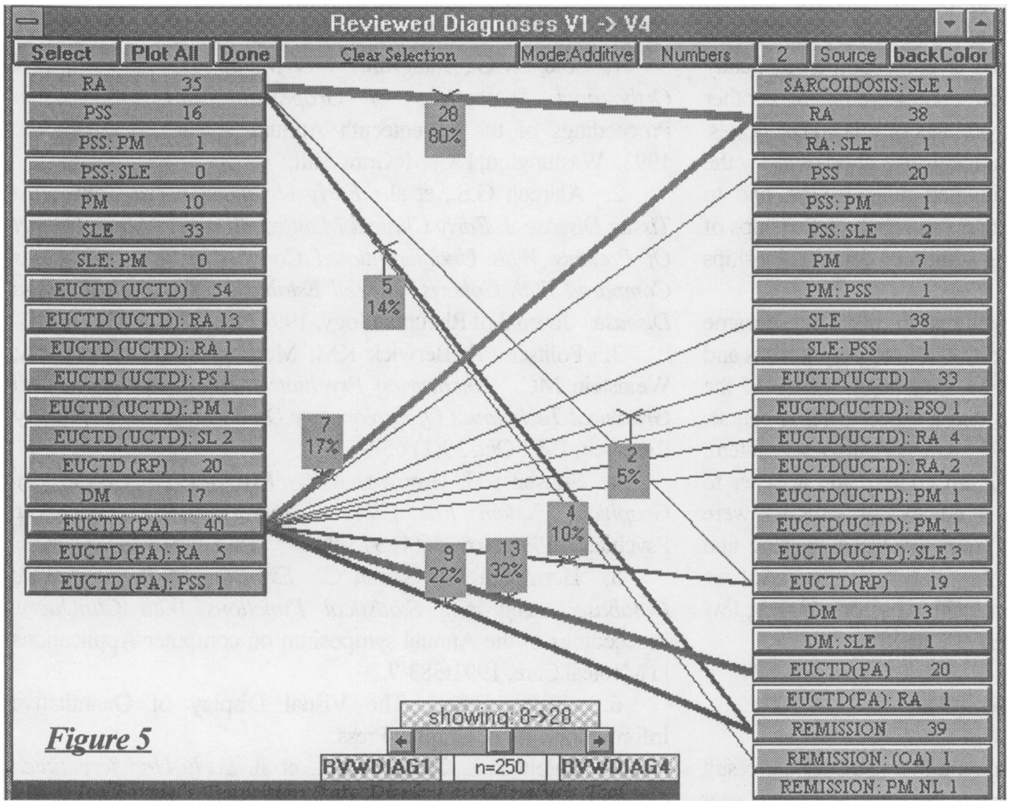
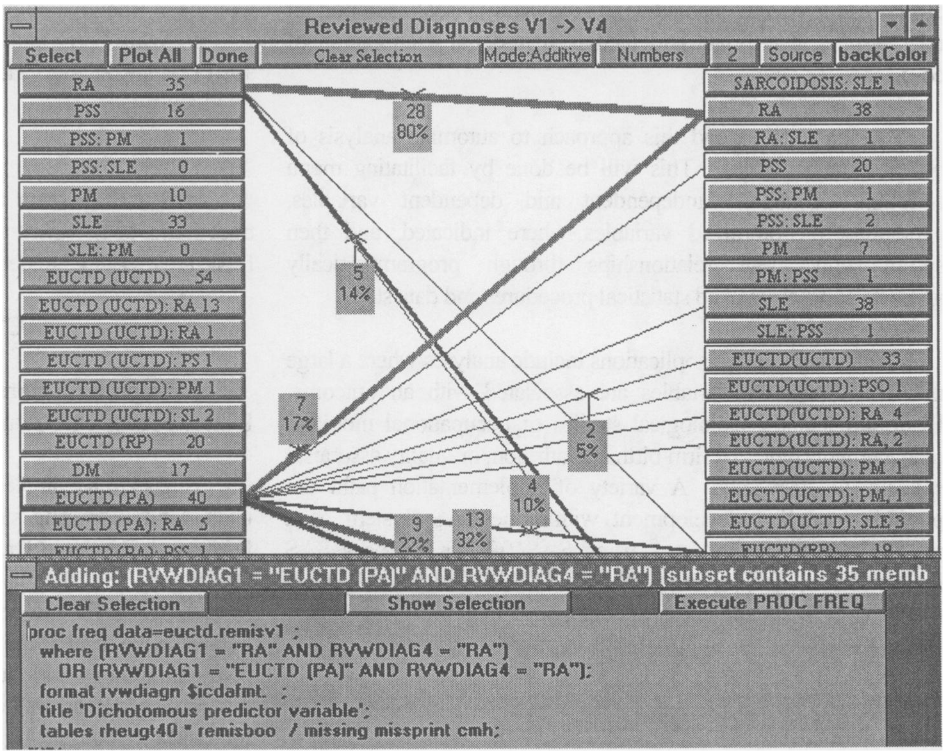


Figure 5

the data to its most coarse features. Similarly, the threshold can be incrementally decreased to reveal the full granularity.

This technique also supports the direct visual comparison of multiple groups simultaneously. For example, the behavior of populations expected to have comparable distributions is shown in **figure 5**. Although it was anticipated that the Rheumatoid Arthritis (RA) <top left> and Early Undifferentiated Connective Tissue Disease/PolyArthritis (EUCTD(PA)) <bottom left> groups would have a similar distribution, it is visually apparent that the EUCTD(PA) group showed a considerably more variable course. Although the majority (80%) of the patients initially diagnosed with RA remained with RA as their final diagnosis, the EUCTD(PA) patients did not have nearly as predictable a course. By overlying data in this way, it is also possible to direct the computer to lump together groups that have similar characteristics for further analysis. This was implemented for SAS and is shown in **figure 6**. By visually selecting the subgroups of interest, a SAS program (procedure) is constructed by Visual Basic. A similar technique could have generated structured query language (SQL) code for a relational database. The size of the resultant group, in this case 35 patients, is shown in the title bar of the window where the procedure appears. You will also note that there is a button labeled "Show Selection" which will show just the selected transition paths. This subsetting can be built up interactively to include any combinations of subgroups, including groups not currently displayed.



Adding: (RVWDIAG1 = "EUCTD (PA)" AND RVWDIAG4 = "RA") (subset contains 35 memb

```
proc freq data=euctd.remisv1 ;
  where (RVWDIAG1 = "RA" AND RVWDIAG4 = "RA")
    OR (RVWDIAG1 = "EUCTD (PA)" AND RVWDIAG4 = "RA");
  format rvwdiagn $icdafmt.  ;
  title 'Dichotomous predictor variable';
  tables rheugt40 * remisboo  / missing missprint cmh;
```

Figure 6

Finally, this tool has proven to be an effective way to communicate the contained relationships to others. We therefore added the ability to relocate displayed elements and dynamically connect the transition paths, to allow demonstration of another dimension of relationships such as groupings of related categories. Techniques like this, as well as the judicious use of color allow the user to selectively focus on the hilighted relationships, and to re-frame the view by changing the order or spatial relationships of elements. This is useful in identifying additional data relationships and for teaching purposes.

By using this tool, several patterns in our data became apparent. The remission rates for undifferentiated polyarthritis and rheumatoid arthritis were not different, although many of the undifferentiated polyarthritis patients had a milder disease course. These findings are shown in **figure 5**. The Visual Basic system, by writing the appropriate SAS program code, made it easier to find the correlates of disease course. Predictors of remission were determined by selecting subgroups from the visual display and conducting statistical tests for symptoms and laboratory values. The resulting predictors of remission included seronegativity, low sedimentation rate and low tender joint counts[8].

CONCLUSION

Significant improvements in quantitative data display result from visual representations achieved through graphical user interfaces. The use of custom graphics to visualize cross tabulation tables can augment existing tools such as statistics packages, speed interpretation of data, and facilitate presentation of complex analytical results. Prior techniques for data representation did not convert co-occurrence information into an interactive graphical display. Our system enabled us to graphically visualize important relationships in the data and facilitated in-depth examination of co-occurrence data.

We hope to extend this approach to automate analysis of independent variables. This will be done by facilitating menu selection of logical independent and dependent variables, generating dichotomized variables where indicated, and then summarizing these relationships through programmatically generated and dispatched statistical procedures and data steps.

Future substantive applications include analyses where a large number of discrete variables are associated with an outcome. Examples include sociological studies of occupational mobility, management studies of firm birth and attrition, or quality assurance studies of defect data. A variety of implementation paths are possible including development within the SAS System as a user-defined procedure (e.g., using SAS/TOOLKIT), as a SAS application (modeled after SAS/ASSIST which features pull-down menus with "point and click" options), or entirely in Visual Basic.

REFERENCES

1. Cole, W.G., *Integrality and Meaning: Essential and Orthogonal Dimensions of Graphical Data Display.* in Proceedings of the Seventeenth Annual SCAMC Convention. 1993. Washington, DC. McGraw Hill.

2. Alarcon G.S., et al. *Early Undifferentiated Connective Tissue Disease. I. Early Clinical Manifestation In A Large Cohort Of Patients With Undifferentiated Connective Tissue Diseases Compared With Cohorts Of Well Established Connective Tissue Disease.* Journal of Rheumatology, 1991 Sep, 18(9):1332-9.

3. Politser PE; Berwick KM; Murphy JM; Goldman PA; Weinstein MC. *Uncovering Psychiatric Test Information With Graphical Techniques Of Exploratory Data Analysis.* Psychiatry Research, 1991 Oct, 39(1):65-9.

4. Stinson CH; Horowitz MJ. *Psyclops: An Exploratory Graphical System For Clinical Research And Education.* Psychiatry, 1993 Nov, 56(4):375-89.

5. Herrmann FR; Safran C. *Exploring A Hospital-Wide Database: Integrating Statistical Functions With ClinQuery.* Proceedings of the Annual symposium on computer Applications in Medical Care, 1991:583-7.

6. Tufte, E.R., The Visual Display of Quantitative Information. 1983, Graphic Press.

7. Bulpitt K.J., Clements P.J., et al. *Early Undifferentiated Connective Tissue Disease: Iii. Outcome And Prognostic Indicators In Early Scleroderma (Systemic Sclerosis).* Annals of Internal Medicine, 1993 Apr 15, 118(8):602-9.

8. Bormel, JI, Clements P, Paulus, H, Ferguson, L. *Early Undifferentiated Connective Tissue Disease (Euctd) Study: Remissions In Patients Presenting With Polyarthritis.* American College of Rheumatology 57th Annual Scientific Meeting. 1993.

9. Wall, Larry and Randal L. Schwartz, Programming Perl, 1991 O'Reilly & Associates.

TRADEMARKS

ACKNOWLEDGMENTS

Dr. Bormel is currently a fellow supported by the National Library of Medicine (grant LM07092-01).

Thanks to Kevin Nechodom at the CSSRD in Salt Lake City, Utah, for unwavering support, encouragement, and constructive feedback. Thanks to Dr. Douglas S. Bell of the Design Systems Group at the Brigham and Womens Hospital, Boston, Massachusetts for the concept of threshold views which greatly improved the display capabilities. Thanks to the UCLA Division of Rheumatology and the CSSRD for the opportunity to develop this project, and to the Laboratory of Computer Science at MGH for assistance completing this paper.